PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 :
G06F 13/00, 17/30

A1

(11) International Publication Number: WO 98/02820

(43) International Publication Date: 22 January 1998 (22.01.98)

(21) International Application Number: PCT/US97/12093

(22) International Filing Date: 11 July 1997 (11.07.97)

(30) Priority Data:
08/680,914 12 July 1996 (12.07.96) US

(71) Applicant: BITSOURCE, INC. [US/US]; 5150 W. El Camino Real, Los Altos, CA 94022 (US).

(72) Inventor: GOLDMAN, Jonathan; 324 Hedge Road, Menlo Park, CA 94025 (US).

(74) Agent: GOTLIEB, Charles, E.; Law Offices of Charles E. Gotlieb, Suite 300, 540 University Avenue, Palo Alto, CA 94301 (US).

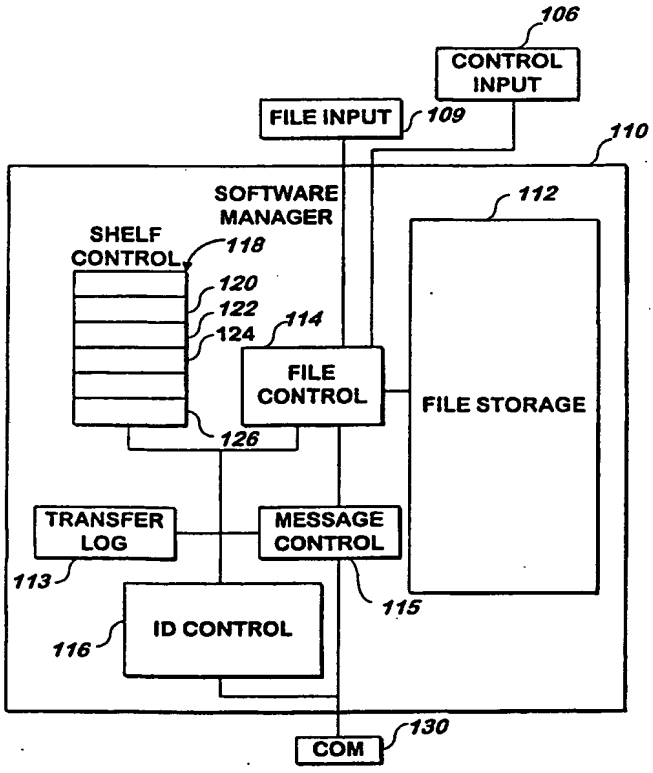
(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: METHOD, APPARATUS AND COMPUTER PROGRAM PRODUCTS FOR EFFICIENTLY DISTRIBUTING INFORMATION

(57) Abstract

Method, apparatus and computer program products duplicate sets of information such as application software as requested by a user. A number of sets of information are stored locally in a file storage (112) for duplication or transmission, and additional sets of information are available for remote retrieval for duplication or transmission.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

METHOD, APPARATUS AND COMPUTER PROGRAM PRODUCTS FOR EFFICIENTLY DISTRIBUTING INFORMATION

Field of Invention

The present invention relates to information distribution systems, and more specifically to intelligent information distribution systems.

Background of the Invention

Information may be distributed from the manufacturer of the information to an end user of the information using a retailer. For example, computer software, one type of information, may be distributed by the manufacturer of the computer software to an end user of the program via a retailer who displays the software on his shelves in a store. A retailer can provide services such as personal pre-sale assistance and may offer a selection of products based upon knowledge of the local marketplace not possessed by the manufacturer.

Some conventional information distribution schemes attempt to bypass the retailer and allow the distribution of information such as computer software directly from the manufacturer to the end user. Distribution of computer software from manufacturers to end users via the Internet is one example of such a scheme. The justification for these schemes is that stacks of shelving in large stores holding physical media which contain the information is an inefficient use of real estate and capital when information such as computer software may be sent to an end user electronically. Additionally, new versions of computer software containing bug fixes and/or new functionality are made instantly available to end users instead of being made available only after the existing physical inventory is

depleted. However, because such schemes can bypass the retailer altogether, they eliminate the services provided by the conventional retailer, and the end user may suffer.

It is possible to include the retailer in the distribution arrangement, while providing much of the efficiency of electronic distribution of information. For example, the retailer may request information to be sent electronically from a manufacturer after consultation with the end user, and the retailer may place the information on a physical medium such as a CD-ROM disc for use by the end user. This allows the retailer to provide an end user a large selection of information without storing all of it in a large retail store, eliminating the carrying costs and much of the real estate costs associated with a conventional retailer. Such a distribution system may allow a retailer to offer a larger selection to end users than would be otherwise economically feasible where information is packaged and stored on a shelf.

Where the amount of information requested by the retailer is large, the time required to receive the information may be unacceptably long due to communication delays. For example, if an end user requests several computer software titles averaging five megabytes each, downloading can take several minutes or more, and the end user may not wish to wait that period of time. An alternative arrangement, whereby each retailer stores all of the available information on a computer storage media such as a hard disk, and then generates the information requested by an end user on a second storage medium such as a CD-ROM may require the retailer to choose between offering a large amount of information at an unacceptably high storage cost, and offering an unacceptably small amount of information compared with the requirements of the end users who visit the store.

Summary of Invention

The present invention stores a subset of information at each retailer, while allowing the retailer electronic access to a much larger amount of information stored at a site shared by many retailers. The subset of information stored at each retailer may be information most likely requested by end users at that retailer, eliminating the communication delays for most end users. The shared site allows each retailer access to a vast amount of information while spreading the costs of storage among many retailers. A mechanism for updating the information as necessary allows a retailer access to the latest version of the information.

A Brief Description of the Drawings

Figure **1A** is a block schematic diagram of a system which stores and provides information according to one embodiment of the present invention.

Figure **1B** is a block schematic diagram of a software manager according to one embodiment of the present invention.

Figure **1C** is a block schematic diagram of a software shelf according to one embodiment of the present invention.

Figure **2** is a block schematic diagram of a set of database entries according to one embodiment of the present invention.

Figure **3** is a flowchart illustrating a method of providing from a first location such as a software manager sets of information necessary to complete a list of several requested sets of information received according to one embodiment of the present invention.

Figure **4** is a flowchart illustrating a method of providing sets of information at a second location such as a

software shelf according to one embodiment of the present invention.

Figure **5** is a block schematic diagram of a conventional computer system.

Detailed Description of a Preferred Embodiment

Sets of information are any logical or physical grouping of information. Information may be represented electronically, which allows for storage using conventional media such as memories, diskettes, hard disks, CD-ROMs, or tapes. Sets of information may be arranged in files of software, and such an arrangement is described below, although other arrangements of sets of information may also be used.

Referring now to Figure **1A**, one embodiment of a system which stores sets of information according to one embodiment of the present invention is shown. Software manager **110** operates as a type of software file "warehouse", having the capacity to store and distribute a wide variety of software, and one or more software shelves **108** operates as a "retail shelf" holding some of the software files stored in the Software Manager **110**. Figure **1A** shows one embodiment of a system containing one software manager **110** and three software shelves **108**, but any number of software managers **110** and software shelves **160** may be used in other embodiments.

Software manager **110** communicates via communication driver **130** across communication lines **100** to one or more communication drivers **132** coupled to one or more software shelves **108**. Communications drivers **130**, **132** may be any form of data communications equipment, such as telecommunications equipment such as conventional modems or local area network interfaces, transmission equipment such as microwave links,

or other forms of communications equipment. Communication lines **100** may be any conventional form of communications, such as dial-up telephone lines, leased telephone lines, microwave communications or others.

In one embodiment, software manager **110** is at least ten miles from at least one software shelf **108**. In other embodiments, software manager **110** is not more than ten miles from all software shelves **108**.

Referring now to Figure **1B**, one embodiment of a software shelf is shown. The software shelf **108** maintains a local set of files in a file store **146** to provide rapid output of these files to an output device **148** upon demand as described below. In one embodiment, the output **148** can record one or more files on a tangible device which is local to the software shelf **108**. The local tangible device may be a CD-ROM disc, if output **148** is a CD-R recordable CD-ROM drive or CD-E erasable CD-ROM drive, or a diskette or tape if output **148** is a conventional diskette or tape drive. The local tangible device may be a printed page if output **148** is a printer. In other embodiments, output device may be a communications device such as a modem, a local area network interface or transmission interface which transmits information to another device for immediate use or for recording in a storage medium not local to the software shelf **108**. In such embodiments, files may be sent by software shelf directly into an end user's personal computer without the need to physically transfer a media from the software shelf **108** to the personal computer.

In one embodiment, software is requested for output using an input device **140**, such as a conventional keyboard and a conventional display monitor, using an indicator such as a catalog number or title, and one or more indicators

corresponding to a request may be stored in requested file storage **142**. Where there is no requested file storage **142**, the indicators are sent as they are entered.

Communications manager **134** formats messages for sending and routes messages received within the software shelf **108**.

In one embodiment, the requested file indicators are sent to a space manager **144**, which receives from software manager **110** and stores and maintains a list of all files stored in file store **146** as well as indicators such as file name and file size, and information related to the location of each file on file store **146**. Space manager **144** compares the indicators requested against a list of the files stored in local file store **146** and if all of the requested files having indicators stored in requested file store **142** are in local file store **146**, those files are sent to the output **148** via an output manager **138** that controls the output process, otherwise the files having indicators stored in the requested file storage **142** that are not in the local file storage **146** are requested as described below.

Referring now to Figures **1A** and **1B**, in another embodiment, all file requests are routed by communications manager **132** through the software manager **110** as described below before checking the contents of the file store **146**. When all file requests are routed through the software manager **110**, the software manager **110** can verify that the most current version of the software requested is available at the software shelf **108**, and can record the transaction in a location secure from the user of the software shelf **108** to allow for tamper-resistant billing for the duplication of all of the requested files.

Referring now to Figures **1A**, **1B** and **1C**, in one embodiment, a software shelf **108** communicates with a software manager **110** using an ID number to allow the software manager **110** to verify the identity of the software shelf **108**, and/or the software shelf **108** to verify the identity of the software manager **110**. The ID control **116**, **136** of the sending device **110**, **160** sends an identification number to the ID control **116**, **136** of the receiving device **110**, **160** which checks the ID number received against a list of valid ID numbers prestored in the ID control **116**, **136** of the receiving device. If ID control **116** verifies the ID number, ID control **116** enables message control **115**, which receives and interprets the remainder of the communications received, and formats the communications sent.

Referring again to Figures **1A** and **1B**, in one embodiment, the software shelf **108** sends the software manager **110** a list of all requested files stored in requested file store **142** after accepting a set of requested files. The list of files in requested file store **142** may correspond to an order from a customer, or may correspond to files which a retailer believes it should "stock" in the local file store **146** of the software shelf **108**. For example, a retailer that intends to offer a special promotional price the next day may wish to stock the software file in anticipation of strong demand. Using input **140**, the retailer can identify whether the file should be stored on file store **146** and written using output device **148** after receipt or merely stored on file store **146** in anticipation of strong demand. Some of the files in the requested file store **142** may be indicated using input **140** as files to "hold" as described below.

Referring now to Figures **1A**, **1B** and **1C**, in one embodiment, after software shelf **108** sends the list of

requested files stored in requested file store **142** to software manager **110**, software manager **110** processes the list as described below. Figure **1C** shows one embodiment of the software manager **110** of Figure **1A**. After ID control **116** verifies the identity and authenticity of the software shelf **108** with whom it is communicating as described above, message control **115** assembles the list of requested files received and queries shelf control **118** to determine which files are already stored in the software shelf **108**. Each portion **120**, **122**, **124**, **126** of shelf control **118** holds a list of the software files stored in the file store **146** on each software shelf **108** authorized to communicate with the software manager **110**. Shelf control **118** maintains these portions **120**, **122**, **124**, **126** to allow software manager **110** to determine which files are already on the software shelf **108** and which files will need to be transferred from the file storage **112** of the software manager **110** to the file store **146** of the software shelf **108** in order to fill the request sent by the software shelf **108**. Shelf control **118** also maintains for each software shelf **108** indicators for the usable size of the file store **146** and can identify the files which are least likely to be used in the future.

Referring momentarily to Figures **1A** and **2**, one embodiment of the arrangement of a list of file information in each portion **120**, **122**, **124**, **126** of one embodiment of shelf control **118** is shown. The file identifiers **220**, **222**, **224**, **226**, **228** such as a name, identifying code and/or a location code which are stored with the size **230**, **232**, **234**, **236**, **238** in bytes corresponding to the file in order of usage. When a file is requested by a software shelf **108** to be written to its output device, the identifiers for that file, such as an identifier **220** such as a file name or a location indicator, and size **230**, as well as hold **240**, **242**, **244**, **246**, **248** and

version **250**, **252**, **254**, **256**, **258** indicators described below, are moved to the top of the list **214**. "Hold" indicators **240**, **242**, **244**, **246**, **248** contains a bit which is set if the file should not be removed from the software shelf, such as if the file was requested in anticipation of large demand as described above, and is not set otherwise. In some embodiments, hold indicators **240**, **242**, **244**, **246**, **248** can expire after a period of time and are cleared . In one embodiment, hold bits **240**, **242**, **244**, **246**, **248** will expire after a number of days have elapsed since the file was requested, the number of days being indicated at the time the file is requested to be held. In another embodiment, hold bits expire if the file information drops below a certain threshold level on the list **214** as indicated at the time the file is indicated held, allowing files to be held but only if the demand for them is maintained above a certain level on the list **214**.

Referring again to Figures **1A**, **1B** and **1C**, file control **114** compares the list of requested files received from the software shelf **108** to the list of files shelf control **118** believes are stored on the software shelf and communicates to the message control **115** the list of indicators of the files already on the software shelf **108** and the list of "missing" files. In one embodiment, the indicators of the files already on the software shelf **108** are the locations on the software shelf. Missing files are the files that are not stored in the file store **146** of the software shelf **108** that must be transferred to the software shelf **108** to fill the request.

In addition, file identifiers corresponding to files which are to be deleted from the file store **146** of the shelf **108** to make room for files which must be sent from software manager **110** to software shelf **108** to fill the order are

identified. In one embodiment, message control **115** identifies the size of all of the files which must be transferred to fill the order as described above, and communicates this information to the shelf control **118** which maintains the size of the file store **146** for each software shelf **108**, and identifies any free space on the file store **146** of the software shelf **108**, and if the free space is insufficient to store the missing files, the shelf control **118** identifies the files which should be deleted from the file store **146** of the software shelf **108**.

In one embodiment, the size of the file store **146** on the software shelf **108** is not maintained in the shelf control **118**, but is instead sent by the software shelf **108** along with the indicators of the requested files. In this embodiment, communications manager **134** queries the space manager **144** which maintains the size of the file store **146** whenever a request is transmitted. This embodiment allows the size of the file store **146** to be adjusted at any time without sending the changed size to the shelf control **118** which may be required where the shelf control **118** stores the size of the file store **146**.

In one embodiment, the least recently used files will be identified for deletion, starting with the least recently used file, the second least recently used file and so on until enough files are deleted to free the space for the files which must be transferred to fill the request. In another embodiment, the sizes of the a certain number, such as ten, of the least recently used files are scanned by shelf control **118** to identify the file having the largest size, which is selected for deletion and removed from consideration, and the process is repeated until enough space for the files which must be transferred to fill the request has been identified. In another embodiment, message control

115 sends the size of the missing files to space manager 144, which performs the determination of which files should be deleted to accommodate the missing files as described above, and deletes the files from the file store 146. In another embodiment, message control 115 sends the size of the missing files to space manager 144 upon receipt of the indicators such as the name and file size of the missing files from the file control 114 for such determination. Space manager 144 then sends indicators such as a file identifier to the shelf control 118 to synchronize the files stored in the file store 146 with the list of files in the shelf control 118.

The list of identifiers of files on the file store 146, the list of identifiers of files not on the file store 146 and the list of identifiers of files which are to be deleted from the file store 146 to make room for the files to be sent to the file store 146 are assembled by message control 115 and sent to the software shelf 108 which sent the request via communications drivers 130, 132. Communications manager 134 receives the lists and routes the list of files stored to space manager 144. Space manager 144 maintains its own list of the files stored in file store 146 and confirms that the list of files stored on the file store 146 is accurate and that all of the files to be deleted are located on the file store 146, by sending a confirmation message to the message control 115 in the software manager 110, and the confirmation is relayed by communications manager 134 to message control 115. Space manager 144 then deletes the files which are indicated for deletion. In one embodiment, files are deleted by removing entries related to them in a file allocation table and the list of files stored in the file store 146 maintained in space manager 144 and marking the space in the file store 146 which is occupied by the files as available for reuse. The files which are not located on the file store

146 are retrieved by file control 114 and sent to the file store 146 via space manager 144, which updates its list of files stored on the file store 146. In another embodiment, the files bypass space manager 144 and are sent directly to file store 146 upon receipt by communications manager 134.

Any form of file transfer protocol will operate to transfer the files. In one embodiment, a file transfer protocol is used that can recover from interruptions in the communications between the message control and the communications manager, such as interference in the communications line between the two communications controllers 130, 132 or a power failure at the software manager 110 or software shelf 108. One embodiment, a file transfer protocol breaks up files into blocks sends blocks one at a time using a error correcting code such as a Hamming code, and allows for periodic acknowledgment of receipt of each block by the communications manager 134 in the software shelf 108 to the message control 115 in the software manager 110. Should an acknowledgment not be received, communications resume with the sending of the block following the block for which a valid acknowledgment was received by message control 115.

New versions of the software files may be made available to the software shelf 108 by installing them on the software manager 110. In one embodiment, shortly after new versions of the software are input to the software manager 110 as described below, file control 114 directs shelf control 118 to identify all software shelves 108 which presently store the old version, and message control 115 is directed to initiate contact with each of those software shelves 108, direct each of them to delete the old version, and download the new version as if the new version had been requested described above. Referring momentarily to Figure 2, in one

embodiment, the position of the indicator of the file in the list **214** is preserved by replacing the entry for such file with the new version. In another embodiment the file indicator is inserted at the head **220** of the list **214** as if the new version of the file had been requested as described above. In another embodiment, the new versions are not transferred at once, instead, the transfer is made as the file is requested. In one embodiment, this is accomplished by replacing the file indicators for the old version with indicators for the new version, moving the old version to the bottom of the list and inserting a pointer **250, 252, 254, 256, 258** in the list **214** at the position of the old version, the pointer **250, 252, 254, 256, 258** pointing to the new version. When the file is requested, the pointer **250, 252, 254, 256, 258** is checked, and the new version is sent in its place. The old version will be deleted as space requires because its indicators were moved to the bottom of the list. In another embodiment, the old version is marked for deletion using the version indicator **250, 252, 254, 256, 258** and at the time of any file transfer to the shelf, the old version is identified for deletion.

A new file or new version is inserted using file input **109**. The file input **109** may be a conventional CD-ROM drive, a conventional floppy diskette drive, a conventional hard drive, a conventional tape drive, or a modem under control of file control **114**. File input **109** accepts a file and transfers it to file storage **112**. In one embodiment, file input **109** is coupled to file control **114** which transfers the incoming file to file storage **112**. In another embodiment, file input **109** is also coupled directly to file storage **109** allowing for control by file control **114** and direct file transfer to file storage **112**.

In one embodiment, a control input **106** is used to provide to file control **114** file information such as file indicators and version number. In one embodiment, control input **106** is a conventional terminal or conventional x86 IBM-compatible personal computer configured to operate as a terminal. Characteristics such as file size may be entered into control input **106** in one embodiment, or calculated by file control **114** in another embodiment. In still another embodiment, control input **106** is not used, and all required information is accepted by file input **109** along with the file. Identifiers such as those described with respect to list **214** of Figure **2** received from control input **106**, file input **109** or calculated as described above are stored in file control **114** along with information related to their location in file storage **112**.

In one embodiment, files are sent from software manager **110** to software shelf **108** in encrypted format so that the files cannot be copied without requesting them as described above. The encryption may be any conventional encryption technology. In one embodiment, the files are sent and stored in blocks which are out of order. Output manager **138** unencrypts the files prior to sending them to output **148**. For example, where the files are stored on file store **146** in out of sequence blocks, output manager **138** reassembles the blocks in the proper order.

In one embodiment, files which are stored in file store **146** are sent to the output **148** while the missing files are received from the software manager **110**. In other embodiments, all requested files are received at the software shelf **108** prior to sending them to the output **148**.

In one embodiment, indicators of files which are sent to output **148** are logged into transfer log **113** along with an

identifier of the software shelf **108** requesting the file to allow for charging for the file transfer. In one embodiment, the indicator and identifier is written after the file is received by the software shelf **108** to avoid an overcharge which may otherwise occur if the transfer is interrupted and does not complete. In one embodiment, the transfer log **113** is in the software manager **110** as shown in Figure **1C** and which, as described above, receives the list of requested files of requested files **142**. The transfer log **113** is coupled to, and under control of, message control **115** in one embodiment, file control **114** in another embodiment, and both message control **115** and file control **114** in another embodiment. In another embodiment(not shown), transfer log **113** resides in software shelf **108** coupled to, and under control of, output manager **138**.

In one embodiment, certain commands received from input **140** are interpreted by communications manager **134** as a request for a catalog of available titles of all the files available to software shelf **108**, and such a catalog may be displayed on input **140** by communications manager **134** in response to such a command. In one embodiment, communications manager obtains the catalog by requesting from file control **114** a list of indicators corresponding to the files stored in file storage **112**. In one embodiment, shelf control **118** maintains a list, input via control input **106** or file input **109**, of which files in the file storage **112** are available to an individual shelf **108**. In such embodiment, file control **114** verifies the availability to the software shelf **108** of each file in file storage **112** prior to sending an indicator corresponding to that file to the communications manager **132** for display at input **140**. Information such as price may also be entered via input **140** or control input **106** and displayed with the other information in the catalog.

Where pricing is input via control input **106**, it may be stored in shelf control **118** to allow for different pricing for each software shelf **108**.

Referring now to Figure **3**, a method of providing from a first location, such as a software shelf, sets of information necessary to complete a list of several requested sets of information according to one embodiment of the present invention is shown. As described above, a set of information is any logically or physically grouped information, such as a computer file or computer program. A first number of sets of information is stored **306** at a first location. Identifiers of a second number of requested sets of information are received **308** from a second location and examined to identify the requested sets of information not stored at the first location **312**. The identifiers received are optionally stored **310**, for example to facilitate billing. An optional step of determining **314** which sets of information should be deleted from the second location to make room for the files to be transferred in step **318** may be performed, and the resulting list of identifiers of sets of information may be communicated **316** to the second location. A third number of the requested sets of information which are not stored at the first location are transferred from the first location to the second location **318** to complete the request. In one embodiment, the first number of sets of information stored in the first location will be higher than a number of sets of information stored at the second location. Because the first location stores some of the sets of information stored at the second location, in some embodiments, the third number will be less than the second number. However, the first, second and third numbers may be any number in other embodiments.

An optional step of maintaining **320** at the first location a list of identifiers and other information related

to the sets of information stored at the second location is performed by removing from the list the information related to the sets of information communicated for deletion in step **316** and adding to the list information related to the files transferred in step **318**. The list may be used to perform the identification step **312** and the creation step **314**.

Referring now to Figure **4**, a method of providing sets of information at a second location, such as a software shelf, according to one embodiment of the present invention is shown. A catalog containing indicators corresponding to available sets of information is optionally received **408** from a first location, the catalog is displayed **410**, for example to a consumer or retailer, and a list of requested indicators corresponding to the requested sets of information is optionally accepted **412**. The requested sets of information may be the sets of information a consumer or retailer is requesting to have on the second location, the location local to the retailer or consumer, and may request to have duplicated as described below. Alternatively, the information may be requested to be transferred to the second location without an immediate request for duplication in anticipation of strong demand. A list of indicators corresponding to the requested sets of information is communicated **414** to a first location such as a software manager, and a set of indicators describing the sets of information already stored at the second location is optionally received **416** as a confirmation that the corresponding sets of information are believed stored at the second location. A set of indicators corresponding to sets of information stored at the second location that should be deleted to make room for the missing sets of information is optionally received **418** and the sets of information corresponding to the set of indicators are deleted **420** from the second location. In one embodiment, deletion occurs via

marking the entries in a a file allocation table corresponding to space occupied by the sets of information to be deleted as being available for reuse, but the information itself is not physically removed. The requested but missing sets of information not already at the second location are received **422**. Having now received the missing sets of information, the requested sets of information are duplicated, or transmitted for duplication elsewhere **424**. In one embodiment, indicators corresponding to the sets of information requested in step **414** are recorded **426** to allow for charging the owner of the software shelf for the information distributed.

The methods of Figures **3** and **4** may be used to implement the software manager, software shelf pair described above with reference to Figures **1B** and **1C**. It is apparent that, with reference to Figures **3** and **4**, some of the steps that would be performed at the first location may be performed at the second location, and some of the steps performed at the second location may be performed at the first location to achieve the same or similar functions.

Referring now to Figure **1A**, each of the software manager **110** and the software shelf **108** or the methods described above may be implemented in software on a conventional computer system. As shown in Figure **5**, a conventional computer system includes a processor **501** coupled to a first storage device **503** such as a memory, and a second storage device **508** such as a disk storage system. A user may interact with the system via a keyboard **504**, mouse **505** and a monitor **506**. Computer program product reader **507**, such as a memory, hard drive, floppy disk drive or CD-ROM drive can is coupled to processor **501** to allow computer readable program code devices **511** such as encoded bits of a computer program product **511** such as a memory, hard disk, floppy disk, CD-ROM or other storage

device to be input to processor **501** and stored in memory **503** and/or disk storage **508** for use in controlling the operation of the computer system **509** when computer program product **510** is inserted into slot **512** of computer program product reader **507**. Communication interface **513** such as a modem is also coupled to processor **501** to allow for remote communications, such as those described above. An operating system is loaded into memory **503** or disk storage **508** or a combination of memory **503** and disk storage **508** and instructs processor **501** to load and execute computer program products **510** comprising applications such as those which function as a software shelf or software manager, described above. Many conventional applications are distributed on computer program products **510**, such as diskettes, storage devices such as ROMs or disk storage systems, each containing computer readable code devices **511** which cause the computer system **509** to operate in a particular manner such as is described herein when a copy of the code devices is loaded into the computer system **509**.

In one embodiment, the methods and apparati described herein are implemented in software and distributed on a computer program product to operate on a conventional computer system such as the computer system **509** shown in Figure **5**. In one embodiment, the software manager is a conventional industry standard 586 or 686 IBM-compatible server with 40 gigabytes of hard disk storage, and the software shelf is a conventional industry standard 586 IBM-compatible personal computer with 10 gigabytes of hard disk storage. The present average size of a software application program is about ten megabytes. As average software sizes change, the storage capacities may adjust to match the new size of software. Storage sizes may also be adjusted to match the number of sets of information such as software application programs that it is desirable to store.

What is claimed is:

1. An apparatus for providing at least a set of information requested by a user, the information having at least one indicator, comprising:

a selector having an input for receiving from the user at least one of the indicators of the requested sets of information from among a number of sets of information, and an output for transmitting the indicators received at the selector input;

an information storage device for storing less than all of the number of sets of information and having an input and an output;

a communication device having an input for receiving at least one set of requested information and an output for providing at least one requested set of information;

an information output having an input and an output for providing at least one set of the requested information; and

a communications manager having a plurality of inputs coupled to the communication output and the selector output, and a plurality of outputs coupled to the communications input input and the information storage device input, the communications manager for accepting the requested file indicators from the selector, transmitting at least one of the indicators received from the selector to the communication device and receiving at least zero sets of information from the communication device.

2. The apparatus of claim 1 additionally comprising a requested file store having an input coupled to the selector for receiving at least one indicator of at least one requested set of information, the requested file store for storing at least one said indicator and an output coupled to

the communications manager for providing said indicators to communications manager.

3. The apparatus of claim 1 additionally comprising:

an identifier having an output coupled to the communications manager for providing a code to the communications manager, said code useful for identifying the apparatus; and

wherein the communications manager is additionally for providing said code to communications device, and said communications device additionally for communicating said code.

4. The apparatus of claim 1 additionally comprising a space manager having an input coupled to at least one of the communications manager outputs, a first output coupled to at least one communications manager input, and a second output coupled to the information storage device input, the space manager for receiving sets of information from the communications manager and transferring said sets of information to the information storage device and for identifying to the communications manager sets of information stored on the information storage device.

5. An apparatus for providing at least one requested sets of information to at least one remote information storage device comprising:

a communication driver having a first and second input and a first and second output, the communications driver for relaying messages received from the first input to the first output, and for relaying messages from the second input to the second output;

an addressable information storage device having an input and an output, the device for storing at least one set

of information comprising the requested sets of information; and

a controller having a first input coupled to one of the communication driver outputs and an output coupled to the information storage device, the controller for receiving indicators corresponding to the requested sets of information, and causing its output to cause the information storage device to place at its output at least one set of information corresponding to the indicators received that are different from information stored on the remote information storage device.

6. The apparatus of claim 5 wherein the controller comprises:

a remote storage device database having an input and an output for storing a plurality of identifiers corresponding to sets of information located on at least one remote information storage device; and

a remote storage device database manager having a first input coupled to the remote storage device database output, a second input coupled to the controller input and a first output coupled to the remote storage device database input for comparing the set of identifiers received said second input and the identifiers corresponding to sets of information located on the remote storage device.

7. The apparatus of claim 6 wherein the controller additionally comprises an identifier control having an input coupled to the communication input and an output coupled to the remote storage device database manager for receiving a first identifier from the remote information storage device, identifying the remote information storage device using the identifier, and sending to said output a second identifier corresponding to the remote storage device.

8. A method of providing at least one set of requested information having at least one identifier from a first location comprising a first information storage, to a second location comprising a second information storage, the method comprising:

storing a plurality of sets of information comprising the requested information;

receiving at least one first identifier of the at least one requested set of information;

receiving at least one second identifier of at least one set of information different from the at least one requested set of information; and

communicating the at least one requested set of information.

9. The method of claim 8 comprising the additional step of identifying the missing sets of information comprising at least one set of information corresponding to a plurality of all of the identifiers received not stored in the first information storage.

10. The method of claim 9 comprising the additional steps of:

determining at least one size of the missing sets of information; and

communicating to the second location at least one indicator corresponding the at least one size of the missing sets of information.

11. The method of claim 10 wherein the indicators corresponding to the at least one size of the missing sets of information comprise at least one identifier of a set of

information different from the identifiers received and corresponding to sets of information different from the sets of information communicated.

12. The method of claim 11 comprising the additional steps of:

adding to a list at least one indicator corresponding to the sets of information communicated to the second location; and

deleting from the list at least one indicator corresponding to the size of the missing sets of information communicated.

13. The method of claim 12 comprising the additional step of recording at least one of the identifiers received.

14. A method of providing at a second location at least one requested set of information having at least one indicator, the method comprising:

storing a first number of at least a plurality of stored sets of information;

accepting at least one indicator of at least one requested set of information;

requesting from a first location at least one set of information different from the stored sets of information;

receiving the sets of information requested; and

duplicating the requested sets of information.

15. The method of claim 14 wherein the requesting step comprises communicating to a first location the indicator corresponding to the sets of information requested.

16. The method of claim 14 comprising the additional steps of:

communicating to the first location at least one identifier corresponding to at least one set of information stored; and

receiving a confirmation that the sets of information corresponding to the identifiers communicated are stored sets of information.

17. The method of claim 14 comprising the additional step of displaying a plurality of indicators corresponding to the sets of information in the first number and displaying at least one indicator corresponding to at least one set of information not stored at the second location.

18. The method of claim 17 comprising the additional steps of:

requesting from a location different from the first location at least on indicator corresponding to at least one set of information not stored at the second location;

receiving from the location different from the second location at least one indicator corresponding to at least one set of information not stored at the second location; and

displaying at least one indicator corresponding to the sets of information corresponding to the indicators received.

19. The method of claim 14 comprising the additional steps of:

receiving at least one indicator corresponding to sets of information stored at the second location; and

deleting at least one set of information corresponding to the indicators received.

20. A computer program product comprising a computer useable medium having computer readable code embodied therein for providing at least one set of requested information having at least one identifier from a first location comprising a first information storage, to a second location comprising a second information storage, the computer program product comprising:

computer readable code devices configured to cause a computer to store a plurality of sets of information comprising the requested information;

computer readable code devices configured to cause a computer to receive at least one first identifier of the at least one requested set of information;

computer readable code devices configured to cause a computer to receive at least one second identifier of at least one set of information different from the at least one requested set of information; and

computer readable code devices configured to cause a computer to communicate the at least one requested set of information.

21. The computer program product of claim 20 additionally comprising computer readable code devices configured to cause a computer to identify the missing sets of information comprising at least one set of information corresponding to a plurality of all of the identifiers received not stored in the first information storage.

22. The computer program product of claim 20 additionally comprising:

computer readable code devices configured to cause a computer to determine at least one size of the missing sets of information; and

computer readable code devices configured to cause a computer to communicate to the second location at least one indicator corresponding the at least one size of the missing sets of information.

23. The computer program product of claim 22 wherein the indicators corresponding to the at least one size of the missing sets of information comprise at least one identifier of a set of information different from the identifiers received and corresponding to sets of information different from the sets of information communicated.

24. The computer program product of claim 23 additionally comprising:

computer readable code devices configured to cause a computer to add to a list at least one indicator corresponding to the sets of information communicated to the second location; and

computer readable code devices configured to cause a computer to delete from the list at least one indicator corresponding to the size of the missing sets of information communicated.

25. The computer program product of claim 24 additionally comprising computer readable code devices configured to cause a computer to record at least one of the identifiers received.

26. A computer program product comprising a computer useable medium having computer readable code embodied therein for providing at a second location at least one requested set of information having at least one indicator, the computer program product comprising:

computer readable code devices configured to cause a computer to store a first number of at least a plurality of stored sets of information;

computer readable code devices configured to cause a computer to accept at least one indicator of at least one requested set of information;

computer readable code devices configured to cause a computer to request from a first location at least one set of information different from the stored sets of information;

computer readable code devices configured to cause a computer to receive the sets of information requested; and

computer readable code devices configured to cause a computer to duplicate the requested sets of information.

27. The computer program product of claim 26 wherein the computer readable code devices configured to cause a computer to request comprise computer readable code devices configured to cause a computer to communicate to a first location the indicator corresponding to the sets of information requested.

28. The computer program product of claim 26 additionally comprising:

computer readable code devices configured to cause a computer to communicate to the first location at least one identifier corresponding to at least one set of information stored; and

computer readable code devices configured to cause a computer to receive a confirmation that the sets of information corresponding to the identifiers communicated are stored sets of information.

29. The computer program product of claim 26 additionally comprising computer readable code devices configured to cause a computer to display a plurality of indicators corresponding to the sets of information in the first number and displaying at least one indicator corresponding to at least one set of information not stored at the second location.

30. The computer program product of claim 29 additionally comprising:

computer readable code devices configured to cause a computer to request from a location different from the first location at least on indicator corresponding to at least one set of information not stored at the second location;

computer readable code devices configured to cause a computer to receive from the location different from the second location at least one indicator corresponding to at least one set of information not stored at the second location; and

computer readable code devices configured to cause a computer to display at least one indicator corresponding to the sets of information corresponding to the indicators received.

31. The computer program product of claim 26 additionally comprising:

computer readable code devices configured to cause a computer to receive at least one indicator corresponding to sets of information stored at the second location; and

computer readable code devices configured to cause a computer to delete at least one set of information corresponding to the indicators received.

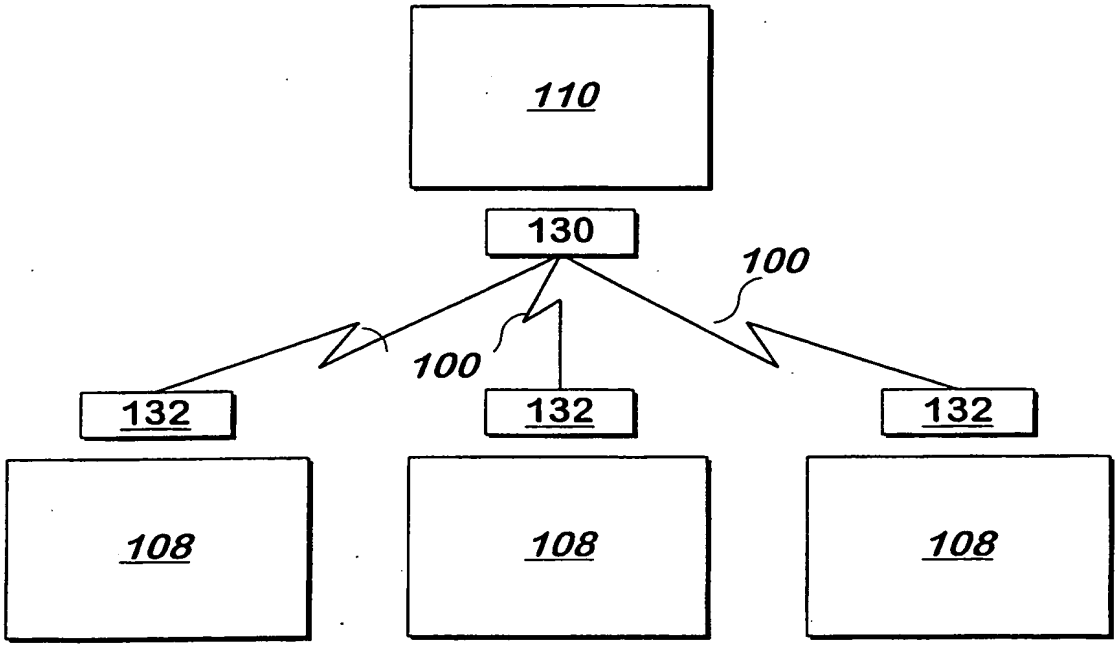


FIG. 1A

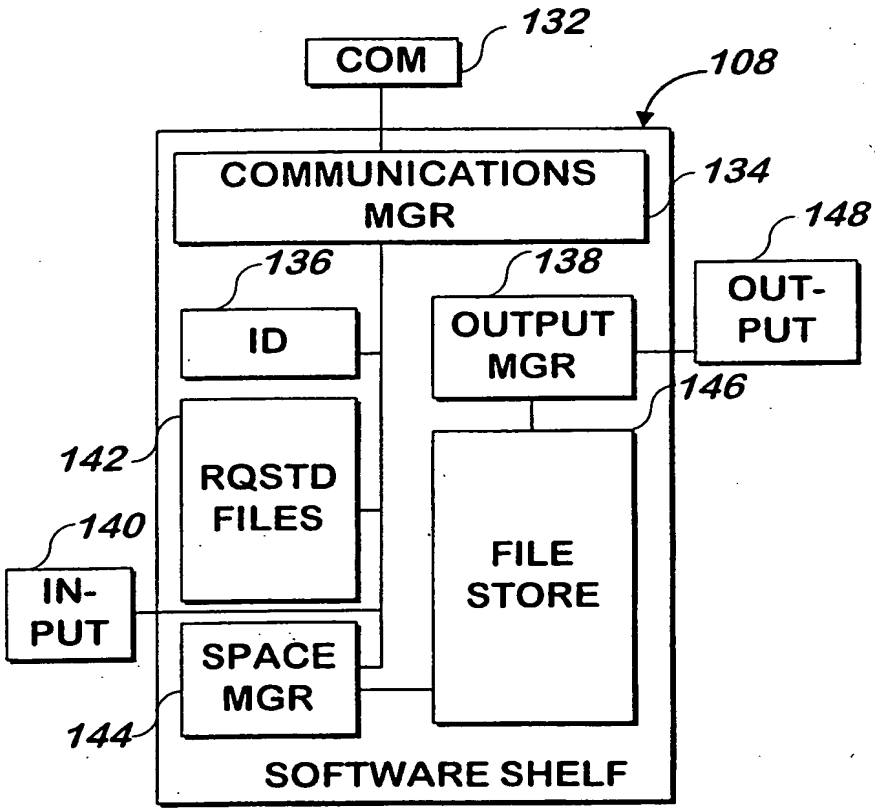

132
COM
108
COMMUNICATIONS MGR
134
148
OUT-PUT
136
ID
138
OUTPUT MGR
146
142
RQSTD FILES
140
IN-PUT
FILE STORE
SPACE MGR
144
SOFTWARE SHELF

FIG. 1B

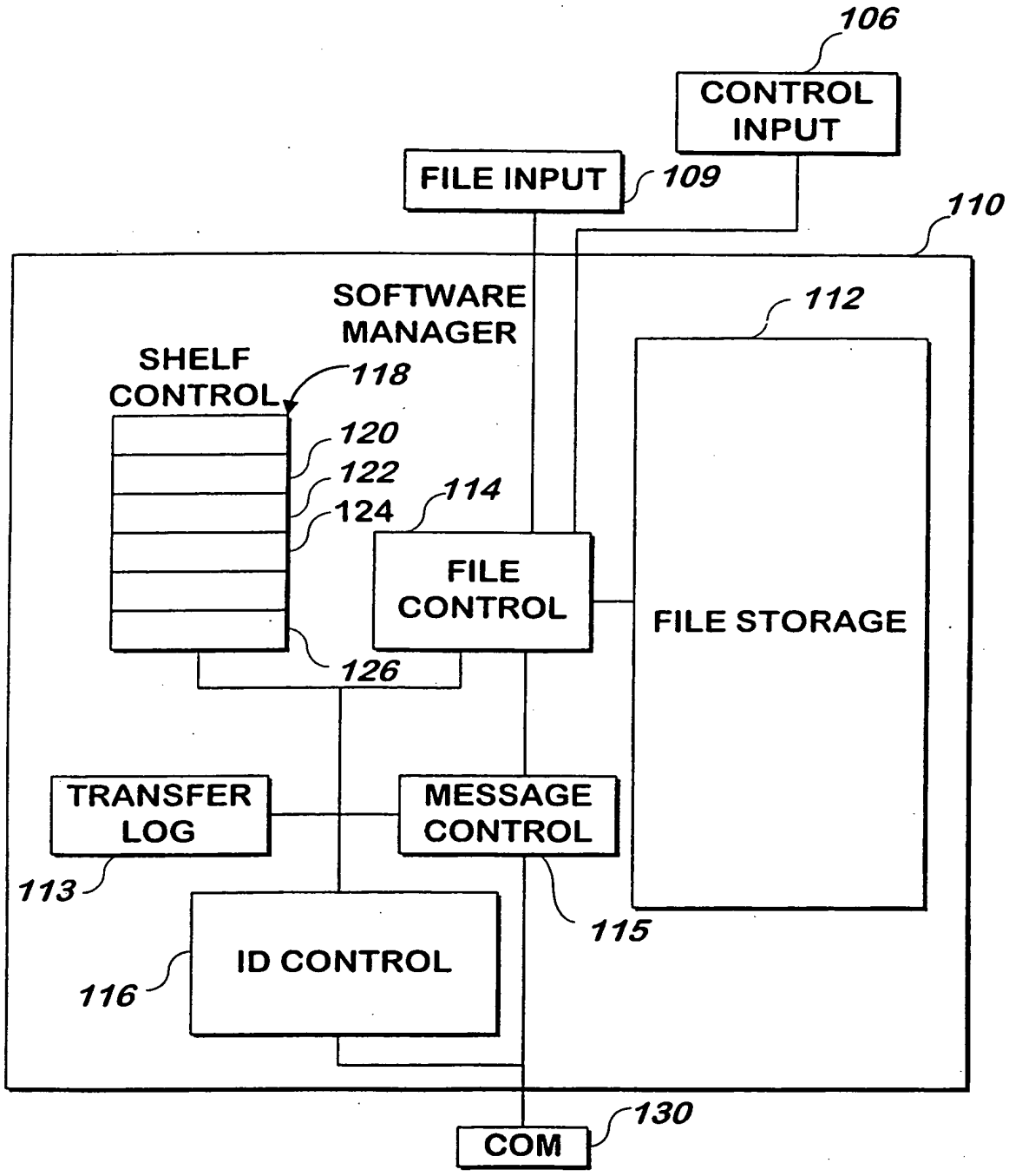

106
CONTROL INPUT
FILE INPUT
109
110
SOFTWARE MANAGER
112
SHELF CONTROL
118
120
122
124
114
FILE CONTROL
FILE STORAGE
126
TRANSFER LOG
MESSAGE CONTROL
113
115
ID CONTROL
116
130
COM

FIG. 1C

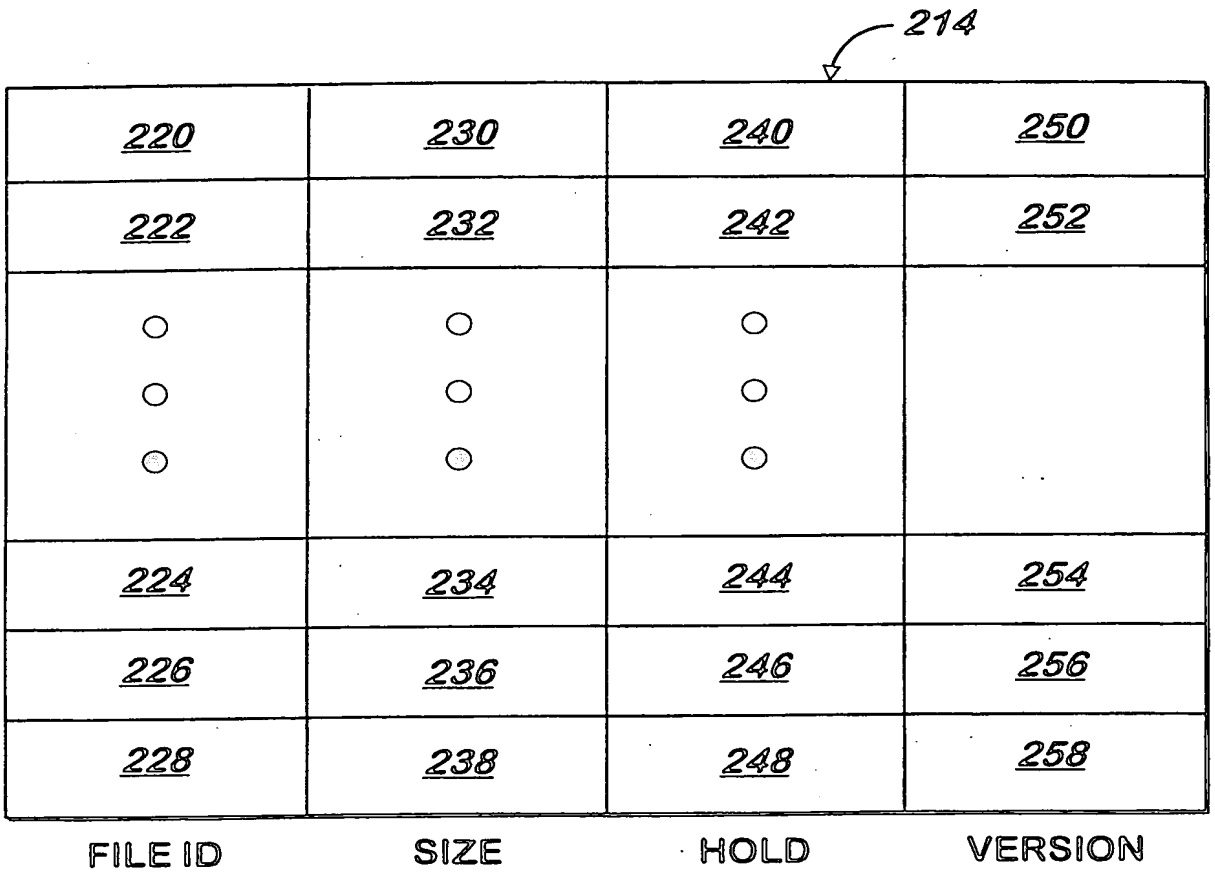

214
220
230
240
250
222
232
242
252
224
234
244
254
226
236
246
256
228
238
248
258
FILE ID
SIZE
HOLD
VERSION

FIG. 2

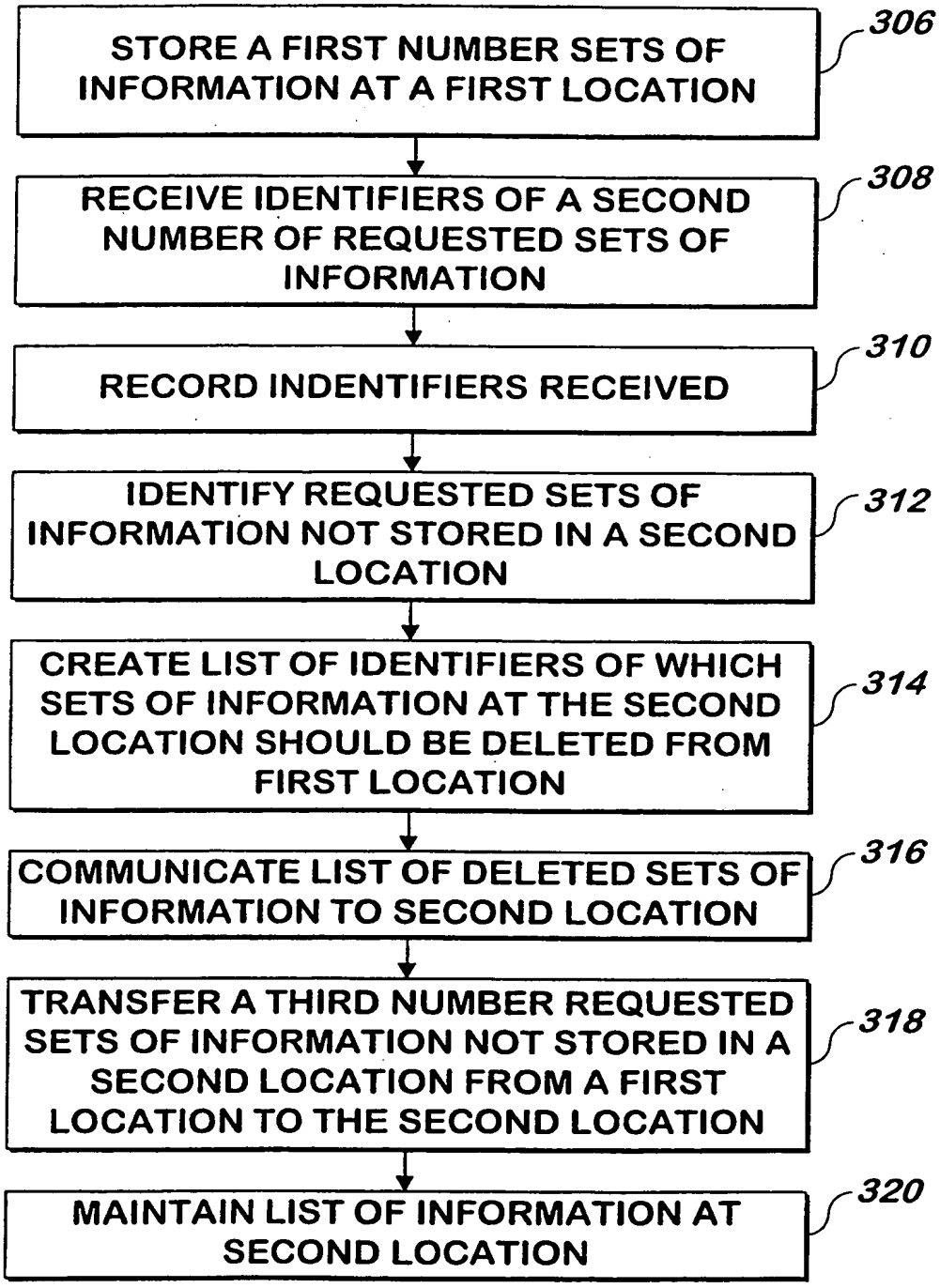

STORE A FIRST NUMBER SETS OF INFORMATION AT A FIRST LOCATION
306
RECEIVE IDENTIFIERS OF A SECOND NUMBER OF REQUESTED SETS OF INFORMATION
308
RECORD INDENTIFIERS RECEIVED
310
IDENTIFY REQUESTED SETS OF INFORMATION NOT STORED IN A SECOND LOCATION
312
CREATE LIST OF IDENTIFIERS OF WHICH SETS OF INFORMATION AT THE SECOND LOCATION SHOULD BE DELETED FROM FIRST LOCATION
314
COMMUNICATE LIST OF DELETED SETS OF INFORMATION TO SECOND LOCATION
316
TRANSFER A THIRD NUMBER REQUESTED SETS OF INFORMATION NOT STORED IN A SECOND LOCATION FROM A FIRST LOCATION TO THE SECOND LOCATION
318
MAINTAIN LIST OF INFORMATION AT SECOND LOCATION
320

FIG. 3

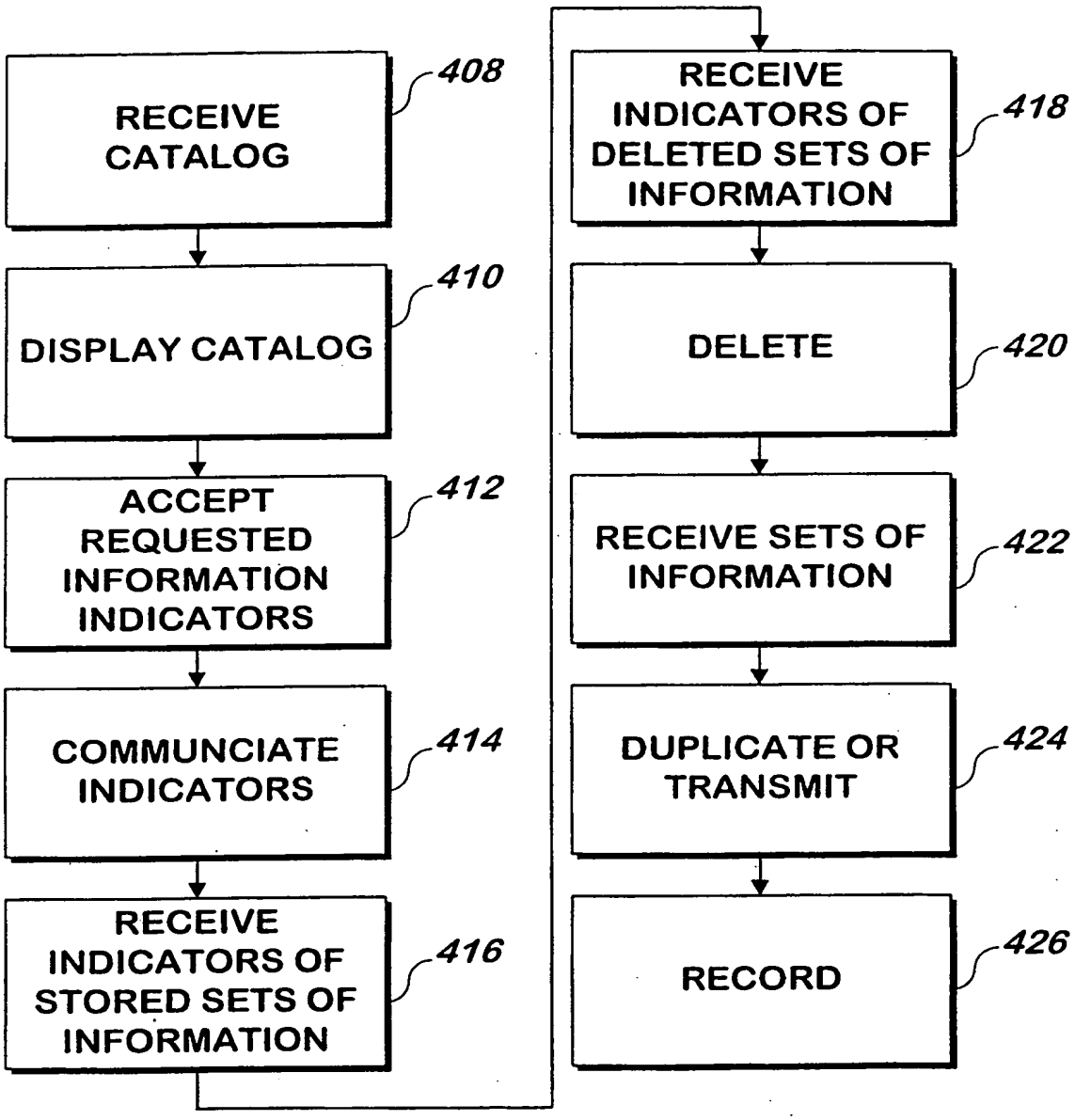

RECEIVE CATALOG
408
DISPLAY CATALOG
410
ACCEPT REQUESTED INFORMATION INDICATORS
412
COMMUNCIATE INDICATORS
414
RECEIVE INDICATORS OF STORED SETS OF INFORMATION
416
RECEIVE INDICATORS OF DELETED SETS OF INFORMATION
418
DELETE
420
RECEIVE SETS OF INFORMATION
422
DUPLICATE OR TRANSMIT
424
RECORD
426

FIG. 4

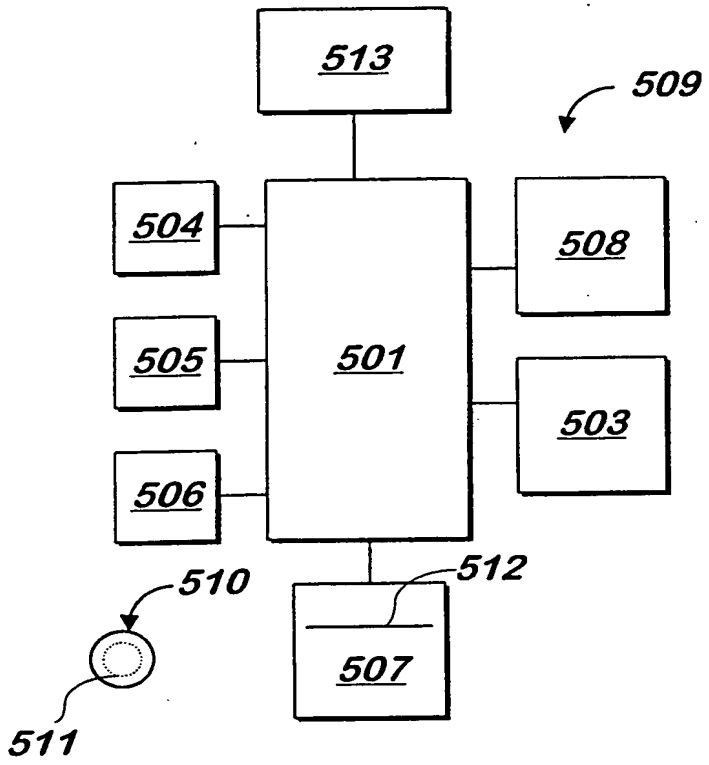


Fig. 5

INTERNATIONAL SEARCH REPORT

Internat[redacted]application No.
PCT/US97/12093

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :G06F 13/00, 17/30
US CL :395/200.31, 610

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 395/200.31, 200.32, 200.33, 200.49, 216, 610; 364/479.01, 479.04

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,267,171 A (SUZUKI ET AL) 30 November 1993, col 2, lines 35-68.	1-31
A	US 5,166,886 A (MOLNAR ET AL) 24 November 1992.	1-31
A	US 4,949,257 A (ORBACH) 14 August 1990.	1-31
Y	US 4,803,614 A (BANBA ET AL) 07 February 1989, cols 1-2.	1-31

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search

06 NOVEMBER 1997

Date of mailing of the international search report

23 DEC 1997

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231

Facsimile No. (703) 305-3230

Authorized officer

VIET VU

Telephone No. (703) 305-9600

Form PCT/ISA/210 (second sheet)(July 1992)★